AD 613719

LINEAR PROGRAMMING IN A MARKOV CHAIN

Philip Wolfe
Computer Sciences Department
The RAND Corporation

and

G. B. Dantzig
Mathematics Department
The RAND Corporation

P-1842

November 23, 1959

LINEAR PROGRAMMING IN A MARKOV CHAIN

Philip Wolfe
Computer Sciences Department
The RAND Corporation

and

G. B. Dantzig
Mathematics Department
The RAND Corporation

P-1842

November 23, 1959

Presented at a Meeting of the American Mathematical Society in New York, April, 1960, and at a Meeting of the Econometric Society in Palo Alto, California, August, 1960

CLEARINGHOUSE FOR FEDERAL SCIENTIFIC AND TECHNICAL INFORMATION, CFSTI
INPUT SECTION 410.11

LIMITATIONS IN REPRODUCTION QUALITY OF TECHNICAL ABSTRACT BULLETIN DOCUMENTS, DEFENSE DOCUMENTATION CENTER (DDC)

AD 613719

☐ 1. AVAILABLE ONLY FOR REFERENCE USE AT DDC FIELD SERVICES. COPY IS NOT AVAILABLE FOR PUBLIC SALE.

☒ 2. AVAILABLE COPY WILL NOT PERMIT FULLY LEGIBLE REPRODUCTION. REPRODUCTION WILL BE MADE IF REQUESTED BY USERS OF DDC.

☒ A. COPY IS AVAILABLE FOR PUBLIC SALE.

☐ B. COPY IS NOT AVAILABLE FOR PUBLIC SALE.

☐ 3. LIMITED NUMBER OF COPIES CONTAINING COLOR OTHER THAN BLACK AND WHITE ARE AVAILABLE UNTIL STOCK IS EXHAUSTED. REPRODUCTIONS WILL BE MADE IN BLACK AND WHITE ONLY.

TSL-121-2 65

DATE PROCESSED: [illegible]

PROCESSOR: [illegible]

SUMMARY

An infinte Markov process with a finite number of states is studied in which the transition probabilities for each state range independently over sets which are either finite or are convex polyhedra. A finite computational procedure is given for choosing those transition probabilities which minimize appropriate functions of the resulting equilibrium probabilities.

CONTENTS

1. INTRODUCTION

Recent studies, cited below, have indicated considerable interest in optimization problems formulable as problems of choosing a set of distributions, constituting the transition probabilities of a finite Markov process, in such a way as to minimize certain "costs" associated with the process.

The following inventory problem is a typical example of this class: Let the n attainable levels of the inventory of an item constitute the n states of a Markov process. Transition from one state to another will occur at the end of each of an infinite sequence of time periods. Owing to the uncertain nature of supply and demand for the item, whose distributions only are assumed known, the effect of a given inventory policy must be described as a distribution. For any inventory policy the probability p_{ij} of transition from state i to state j in one time period is known, as well as the cost c_{ij}, dependent on the policy, which will be incurred if that transition is made. Under any policy the time-series of inventory levels constitutes a Markov process described by the given probabilities. When an initial state for the first period has been given, the long-run probabilities $\bar{p}_{ij}$ are then determined. Intuitively, $\bar{p}_{ij}$ is the probability that, at a typical time period in the indefinite future, the transition from state i to state j will take place. The "long-range expected cost" of using the particular policy is then defined as $\sum_{i,j} c_{ij} \bar{p}_{ij}$.

The computational problem is then that of minimizing this expected cost over all available inventory policies.

The formulation of such a problem as a linear programming problem has been done by Manne [5], d'Epenoux [3], and Oliver [6] for problems in which it is possible to choose the transition probabilities p_{ij}, for each i separately, as one of a given finite set of distributions. The same assumption on the available distributions is made by Howard [4] in his "dynamic programming" treatment of this class of problem. The observation, however, that the problem formulated as a linear program can be efficiently attacked by means of a specialization of the decomposition algorithm for linear programming [2], makes it possible to broaden considerably the class of problems that can be handled, by permitting other descriptions of the sets of available alternatives. In the sequel, two extreme cases are considered: the case described above, on the one hand; and, on the other, the case in which the distributions which may be used are restricted only by being required to satisfy certain linear inequalities. Since these two extreme cases are handled by essentially the same method, intermediate cases, which are of practical interest, can readily be treated by the same technique.

2. THE PROBLEM

Throughout this paper n is a fixed integer. By __distribution__ is meant an n-vector $x = (x_1,..,x_n)$ such that $x_i \geqslant 0$ (all i) and $\sum x_i = 1$.* A Markov process is defined by n distributions $P_i = (p_{i1},..,p_{in})$ for $i = 1,..,n$, where p_{ij} is the probability of transition from state i of the process to state j .

In the problem studied here, a particular Markov process is defined by a choice of distributions from certain sets. In this section and the next, these sets will be assumed finite:

> For each $i = 1,..,n$, let S_i be a finite set of distributions.

In addition, a "cost" $c_i(P)$ is associated with each distribution P in S_i :

> For each $i = 1,..,n$, let c_i be a real-valued function on S_i .

For P in S_i , $c_i(P)$ is thought of as a fee to be paid for the use of the distribution P when passing through state i .

The particular manner in which S_i and c_i are described is not of great importance in the discussion which follows, but it does play an important role in the computational algorithm of Sections 4 and 5. The more extensive discussion of Section 5 can be anticipated by the observation that the algorithm is aimed at handling either of the following two extremes: (a)

---

*The symbol "$\sum$" is used throughout as an abbreviation of "$\sum_{i=1}^{n}$" .

S_i is given as an arbitrary finite set, and c_i as an arbitrary function on S_i ; (b) A finite set of linear inequalities in $n+1$ variables is given, defining an $n+1$-dimensional polyhedron in such a way that the first n coordinates of any point of this polyhedron form a distribution. The first n coordinates of any extreme point of this polyhedron constitute a member P of S_i , with $c_i(P)$ defined as the minimal $n+1^{st}$ coordinate of all extreme points whose first n coordinates constitute P .*

If now particular P_i in S_i are chosen for each i , then a Markov process is defined. Let x be an equilibrium distribution for this process -- that is, a distribution satisfying relation (2.2) below. The "expected cost" of the process per stage, when the equilibrium x obtains, is then

$$(2.1) \qquad \sum c_i(P_i)\, x_i \ .$$

The Markov programming problem is that of choosing the P_i in such a way that this expected cost is minimized.

Formally, the problem is:

Determine P_i in S_i $(i = 1,..,n)$ such that (2.1) is minimized for all x such that

*It will be seen from the discussion of case (b) in Section 5 that the restriction of S_i to extreme points of the polyhedron is unnecessary, since even if all points were admitted, only extreme points would appear in the solution of the problem. This restriction is made because of the convenience of assuming S_i to be finite.

(2.2) $$x_i \geq 0 , \quad \sum x_i = 1 , \quad \text{and} \quad \sum x_i P_i = x .$$

It will be convenient for the sequel to restate this problem in such a way that the equations (2.2) have constant right-hand sides.

For each i, let T_i be the set of all n-vectors

(2.3) $$Q_i = (p_{i1}, \ldots, p_{ii}-1, \ldots, p_{in})$$

for which $(p_{i1}, \ldots, p_{in}) = P_i$ is in S_i, and define $\bar{c}_i$ on T_i by $\bar{c}_i (Q_i) = c_i (P_i)$, using the correspondence given. The problem may then be stated:

Determine Q_i in T_i $(i=1,\ldots,n)$ such that

(2.4) $$\sum \bar{c}_i (Q_i) x_i$$

is minimized for all x such that

(2.5) $$x_i \geq 0 , \quad \sum x_i = 1 , \quad \text{and} \quad \sum x_i Q_i = 0 .$$

It is clear that any solution $x; P_1, \ldots, P_n$ of the problem stated by (2.1) and (2.2) gives a solution $x; Q_1, \ldots, Q_n$ of the problem (2.4, 2.5), and vice versa.

3. FORMULATION AS A LINEAR PROGRAMMING PROBLEM

The problem (2.4, 2.5) will be solved with the devices developed for the "decomposition" of linear programming problems of special structure [2], specialized to the case at hand. The central idea of this approach is the formulation of the problem to be solved as a linear programming problem whose data consist primarily of the coordinates of points of the set T_i . This will be done in this section. For each i , let the K_i points of T_i be Q_i^k for $k=1,\dots,K_i$. As an abbreviation, let $c_{ik} = \bar{c}_i\,(Q_i^k)$ for all i,k . Consider the linear programming problem:

Minimize

$$(3.1) \qquad \sum\sum_{k=1}^{K_i} c_{ik}\, y_{ik}$$

under the constraints

$$(3.2) \qquad y_{ik} \geqslant 0 \ , \quad \sum\sum_{k=1}^{K_i} y_{ik} = 1 \ , \quad \sum\sum_{k=1}^{K_i} y_{ik}\, Q_i^k = 0 \ .$$

Theorem 1 below will show that this problem is equivalent to the problem of the previous section. In general, replacing a discrete problem by a continuous problem in this manner can lead to a solution that is not discrete, but the Lemma below shows that for the problem studied here the solution of the continuous problem is itself sufficiently "discrete" to ensure equivalence: For each i , only a single Q_i^k is actually involved in the solution of the problem (3.1, 3.2).

LEMMA: There is a solution of the problem (3.1, 3.2) with the property that for each i there is at most one k for which $y_{ik} > 0$.

Proof: The coefficients and right-hand side of the linear programming problem (3.1, 3.2) are displayed in the table below, headed by their variables y_{ik} , where p_{ij}^{k} denotes the appropriate component of the distribution P corresponding to Q_i^k .

Table 1

$$
\begin{array}{ccccccccc}
 & y_{11} & y_{12} & y_{21} & y_{22} & \cdots & \cdots & \cdots & \\
 & & & & & & & & \\
(3.3) & \boxed{\begin{array}{ccccccc}
1 & 1 & 1 & 1 & \cdots & \cdots & \cdots \\
p_{11}^{1}-1 & p_{11}^{2}-1 & p_{21}^{1} & p_{21}^{2} & \cdots & \cdots & \cdots \\
p_{12}^{1} & p_{12}^{2} \cdots & p_{22}^{1}-1 & p_{22}^{2}-1 & \cdots & \cdots & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
p_{1n}^{1} & p_{12}^{2} & p_{2n}^{1} & p_{2n}^{2} & \cdots & \cdots & \cdots
\end{array}} & & & & & & & \boxed{\begin{array}{c} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{array}}
\end{array}
$$

It is a basic property of linear programming problems [1] that, when a solution exists, there is a solution having exactly -- say r -- positive components for which the submatrix consisting of those columns of the coefficient matrix associated with the positive components has rank r . For this problem, denote by B the $(n + 1)$ by r submatrix of (3.3) given by that property; the associated solution will be the one whose existence the lemma asserts. (As a matter of fact, the simplex

method solution of this linear programming problem will yield a solution of just this type.)

Let s be the number of rows of B in which may be found an entry of the form $p_{ij}^k - 1$. Excluding the first row, the other $n - s$ rows have only non-negative entries; since their right-hand sides are zero, and their variables y_{ik} positive, these rows must in fact vanish, and B has just $s + 1$ non-vanishing rows. The non-vanishing rows are, however, linearly dependent (the sum of all rows but the first is zero), whence the rank of B is at most s, that is, $s \geqslant r$. Since B has just r columns, it follows that at most one entry of the form $p_{ij}^k - 1$ can be found on any row of B, so that at most one column of (3.3) can be found in B for j given, which proves the lemma.

Theorem: The programming problems (2.4, 2.5) and (3.1, 3.2) are equivalent, their solutions being related in this way:

Given y_{ik} solving (3.1, 3.2) and satisfying the conclusion of the lemma, let for each i

$$\left.\begin{aligned} x_i &= y_{ik} \\ Q_i &= Q_i^k \end{aligned}\right\} \quad \text{where} \quad y_{ik} > 0 \quad \text{for some} \quad k ,$$

$$\left.\begin{aligned} &x_i = 0 \\ &Q_i \quad \text{arbitrary in} \quad S_i \end{aligned}\right\} \quad \text{if} \quad y_{ik} = 0 \quad \text{for all} \quad k .$$

On the other hand, given x_i, Q_i solving (2.4, 2.5), let

$$y_{ik} = \begin{cases} x_i & \text{for } k \text{ such that } Q_i^k = Q_i , \\ 0 & \text{otherwise} . \end{cases}$$

Proof: Obvious.

4. COMPUTATIONAL ALGORITHM -- THE MASTER PROBLEM

The linear programming problem formulated in the last section has only $n+1$ equations, but it has $\sum K_i$ variables, a number which may be very large, and not even known for problems whose data are given implicitly. The revised simplex method [1] is particularly advantageous for problems having many more variables than constraints. The decomposition algorithm uses this efficiency of the revised simplex method by clearly separating the considerations involving the constraints alone from those connected with the handling of the variables. That part of the problem involving the constraints is called the "master problem," and its handling is set forth in this section. That part of the problem involving the variables, called the "subproblem," is dealt with in the next section. It will be seen that the work of treating the master problem consists of little more than the application of the revised simplex method to the Markov programming problem as formulated in Section 3. The general iterative step is given below, followed by the procedures for initiating the iterative process and for passing from the determination of an initial feasible point (Phase One) to the determination of the solution of the problem (Phase Two). (The phenomenon of degeneracy plays the same role in this algorithm as in any linear programming problem, and it will be supposed that standard methods [1] may be relied upon when necessary.)

THE ITERATIVE STEP

At any step in the course of the solution of the problem (3.1, 3.2) by the revised simplex method, there will be at hand some n+1 column vectors $\bar{Q}^1,..,\bar{Q}^{n+1}$ (of length n+1) constituting a "feasible basis"; that is, they are linearly independent, and the right-hand side of the equations (3.3) may be expressed as a non-negative linear combination of them. (The weights in this linear combination, which of course constitute a solution of equations similar to (3.3) deriving their coefficients from the Q^i, are called collectively a "basic feasible point.")

Let the "cost" $\bar{c}^i$ be associated with the column $\bar{Q}^i$, for $i=1,..,n+1$. The "prices," assumed known, associated with this basis are defined to be the components of the n+1-vector $\pi = (\pi_1,..,\pi_{n+1})$ satisfying the relationships $\pi\bar{Q}^i = \bar{c}^i$ $(i=1,..,n+1)$.

One iteration of the simplex method consists of the following steps:

(i) Find a column Q of the matrix (3.3) which, with its associated cost c, satisfies the relation

$$(4.1) \qquad c - \pi Q < 0$$

(commonly the column for which $c - \pi Q$ is minimal is chosen). This is the only point in the revised simplex method at which all the columns -- or, what is the same thing, all the variables -- in the problem come into play. This step forms the "subproblem," whose study is deferred to Section 5.

(ii) If no column satisfying (4.1) can be found, then the current basis is "optimal," and the solutions of the equations (3.3) solve the linear programming problem.

(iii) Otherwise, add the column found to the current basis, and remove one column in such a way (given by the rules of the simplex method) that the remainder still forms a feasible basis; calculate the new prices, and begin again.

PHASE ONE

The algorithm can be started with precisely the same device, called Phase One, used for the general linear programming problem. This device consists in augmenting the problem with $m+1$ "artificial" variables in terms of which an initial feasible basis, and the prices associated with the corresponding initial feasible basis, are readily given. The algorithm can then be applied to the problem of removing the artificial variables. After this has been done, the required starting conditions for the ordinary application of the algorithm are automatically met.

For $i=1,..,n+1$: let y_i be a non-negative variable; let I_i be the i^{th} column of the $n+1$-order identity matrix; and let $c_i = 1$ be the cost associated with the variable y_i . For this phase, replace all the costs c_{ik} of the original problem with zeroes.

Designating $I_1,..,I_{n+1}$ as the initial feasible basis, employ the iterative step above until the linear form $\sum_{i=1}^{n+1} y_i$ has been minimized. (Note that the initial feasible point is

$(y_1,..,y_{n+1}) = (1,0,..,0)$ and that the initial prices are $\pi = (1,1,..,1)$.)

The above process will reduce the form $\sum_{i=1}^{n} y_i$, and hence each y_i separately, to zero. (If it did not, then the equations (3.2) would have no solution, which is impossible.) Owing to the linear dependence of the equations (3.2), some of the starting columns I_i will remain in the feasible basis at the end of Phase One; this can be shown to cause no difficulty in the ensuing process [1].

PHASE TWO

When Phase One is finished, restore the deleted costs c_{ik} to the columns Q_i^k , using these costs from now on in the determination of the prices π . Repeat the iterative step until it terminates in its part (ii).

At termination, associated with each Q_i^k in the final feasible basis is a component of the "feasible point," the weight given Q_i^k in expressing the right-hand side of the equations as a linear combination of the columns of the basis. For $i = 1,..,n$, according to the Theorem of Section 3, there can be no more than one Q_i^k in T_i in the basis having positive weight; thus let

$$x_i = \begin{cases} \text{weight for } Q_i^k \text{, if positive,} \\ 0 \text{, otherwise.} \end{cases}$$

The resulting $(x_1,..,x_n)$ is the solution of the problem (2.2).

5. THE SUBPROBLEM AND PROOF OF TERMINATION

The detailed discussion of part (i) of the iterative step of Section 4, the "pricing out" operation in the ordinary revised simplex method, was deferred to this section. Given the quantities π , it constitutes the problem of determining some column Q and its associated cost c for which

$$(5.1) \qquad c - \pi Q < 0 ,$$

where Q may come from any of the sets T_i . How this is done depends on the nature of the description of the original sets S_i from which the T_i were obtained. Evidently the problem of satisfying (5.1) from among the union of all the T_i may be "decomposed" into n problems, the i^{th} one of which, for $i = 1,\ldots,n$, is that of satisfying (5.1) for Q in T_i . If any of these "subproblems" can be solved, then the stated problem has been solved.

For each $i = 1,\ldots,n$, one of the two "extreme" cases mentioned in Section 2 may obtain. (Some "intermediate" case might also be considered, but this will not be done here.)

(a) S_i is given directly as a finite set of distributions, a cost $c_i(P)$ being associated with each member P of S_i .

(b) There is given a finite set of linear relations

$$(5.2) \qquad g_j(z) \geqslant 0 , \quad j = 1,\ldots,m ,$$

in the n+1 variables $(z_1,\dots,z_{n+1}) = z$, such that if z satisfies (5.2), then $(z_1,\dots,z_n)$ is a distribution; S_i is defined to be the set of all $P = (z_1,\dots,z_n)$ such that for some z_{n+1}, $z = (P;z_{n+1})$ is an extreme point of the set of all z satisfying (5.2); and for P in S_i, $c_i(P)$ is defined to be the smallest value of z_{n+1} for which $(P;z_{n+1})$ is such an extreme point. (The index i has been omitted above; of course, the relations (5.2) may be different for each i, or even absent.)

In the case (a), there is not much to be said. Phrased, via the definition (2.3), in terms of S_i, relation (5.1) urges the selection of P in S_i for which

$$(5.3)\qquad c_i(P) - \pi P + \pi_i < 0 .$$

Such a P will yield through (2.3) a column Q satisfying (5.1).

Case (b) is more interesting, in view of the fact that the extreme points of the polyhedron defined by (5.2) have not been assumed to be available in advance. Replacing P and $c_i(P)$ in (5.3) by their definitions in this case, it is desired to choose $z = (z_1,\dots,z_{n+1})$ under the constraints (5.2) in such a way that

$$(5.4)\qquad z_{n+1} - \sum_{j=1}^{n} \pi_j z_j + \pi_i < 0 .$$

This is nearly a linear programming problem; if the customary procedure for the simplex method, that of making the left-hand

side of (5.3) as small as possible, is followed, then the task is precisely a linear programming problem: under the constraints (5.2), minimize the left-hand side of (5.4). Having performed this minimization, if the result is not negative, it is of no interest; but if it is negative, then the column $Q = (z_1, .., z_i - 1, .., z_n)$ and its cost $c = z_{n+1}$ constructed from the solution of the problem satisfy equation (5.1). Furthermore, Q will be an extreme point of the polyhedron.

The complete solution of the subproblem then goes as follows: For each $i=1,..,n$, attempt to satisfy (5.1) from T_i -- or, equivalently, attempt to satisfy (5.3), or (5.4), from S_i. If this can be done for any i, part (i) of the iterative step of Section 4 can be accomplished. (It is indifferent to the fact of the convergence of procedure, although probably not to its rate, whether or not the i for which (5.1) is minimized is chosen.) If, on the other hand, (5.1) cannot be accomplished for any i, then part (ii) of the iterative step obtains, and the procedure has terminated.

It remains only to show that the algorithm is finite. This follows immediately, however, from the finiteness of the simplex algorithm for linear programming [1]; because, as described in Section 4, this algorithm is precisely the simplex method applied to the linear programming problem (3.1, 3.2). Whether the sets T_i of columns are described in the manner (a) or (b) above, they are finite in number, and the proof is complete.

REFERENCES

1. Dantzig, G. B., A. Orden, and P. Wolfe, "The Generalized Simplex Method for Minimizing a Linear Form under Linear Inequality Constraints," Pacific Journal of Mathematics, Vol. 5, No. 2, June, 1955, pp. 183-195.

2. Dantzig, G. B. and P. Wolfe, "Decomposition Principle for Linear Programs," Operations Research, Vol. 8, No. 1, January-February, 1960, pp. 101-111.

3. d'Epenoux, F., "Sur un Probleme de Production et de Stockage dans L'Aleatoire," Revue Francaise de Recherche Operationnelle (Societe Francaise de Recherche Operationnelle), Vol. 4, No. 1, 1960, pp. 3-16.

4. Howard, R. A., Dynamic Programming and Markov Processes, published jointly by Technology Press of Massachusetts Institute of Technology and Wiley, New York, 1960.

5. Manne, A. S., "Linear Programming and Sequential Decisions," Management Science, Vol. 6, No. 3, April, 1960, pp. 259-267.

6. Oliver, R. M., "A Linear Programming Formulation of Some Markov Decision Processes," presented at a meeting of The Institute of Management Sciences, Monterey, April, 1960.

SOME SUGGESTED TECHNIQUES FOR
DATA SYSTEM DEVELOPMENT

J. D. Little
W. V. Shelton

April 25, 1961

(Revised December 1961)

P-2264-1